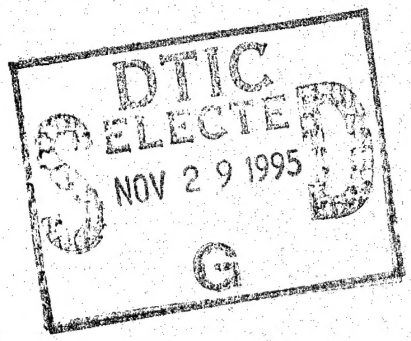
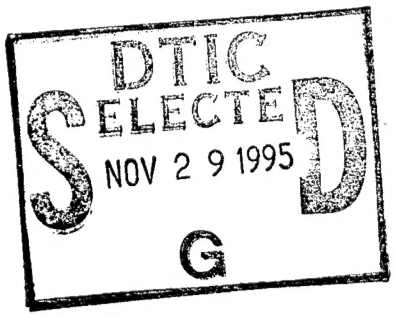
DNA Computing



MITRE

DNA Computing

N. Lewis
P. Weinberger



October 1995

JSR-95-116

Accesion For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

JASON
The MITRE Corporation
7525 Colshire Drive
McLean, Virginia 22102-3481
(703) 883-6997

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	October 1995	

4. TITLE AND SUBTITLE

DNA Computing

5. FUNDING NUMBERS

04-95-8534-01

6. AUTHOR(S)

Nate Lewis, Peter Weinberger

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

The MITRE Corporation
JASON Program Office, Z561
7252 Colshire Drive
McLean, Virginia 22102

8. PERFORMING ORGANIZATION REPORT NUMBER

JSR-95-116

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

ARPA/TIO
3701 North Fairfax Drive, Arlington, Va 22030-1714

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

JSR-95-116

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release, Distribution Unlimited

12b. DISTRIBUTION CODE

Limiter Statement A

13. ABSTRACT *(Maximum 200 words)*

This report examines the potential and limitations of DNA computing. In particular the report examines some of the costs and problems of using DNA computing an large scale problems.

14. SUBJECT TERMS

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Contents

1 DNA COMPUTING

1.1 The Hope

The field of DNA computing has sprung from nowhere to become a fashionable and exciting area. The hope of the field is that the pattern matching and polymerization processes of DNA chemistry, combined with the enormous number of molecules in a pound, will make feasible computations that are now too hard for conventional computers.

1.2 The Basic Approach

The basic approach is that through spontaneous assembly of preselected oligonucleotides (pieces of DNA), DNA strands in a test tube can be used to encode all possible (correct and incorrect) answers to a given problem. The strands are then sorted out according to some defined algorithm (for example, by their length, or by performing various "and", "or", or "not" operations to interrogate the presence of various sequences in an individual DNA strand), with strands that pass the collective requirements deemed to encode the correct answer(s) to the problem. The DNA strands remaining at the end of these various selection steps are amplified using standard biotechnology methods and then are sequenced to obtain the answer(s) to the problem. In principle, only one such strand need remain to be amplified and read out. No arithmetical operations are performed, or have been envisioned, in DNA

computing. Instead, the potential power of DNA computing lies in the ability to prepare and sort through an exhaustive library of all possible answers to problems of a certain size.

1.3 Encoding a Problem onto a DNA Strand

A single strand of DNA can be abstracted as a string made up from the letters A, C, G, T. These letters correspond to the four nucleotide bases found in nature, adenine (A), cytosine (C), guanine (G), and thymine (T). Strings are ordered (first character written on the left, next character to its right, etc). DNA matches this by considering the 5' the left end of the string. Complementary strands of DNA will form a double strand (the famous double helix). Two strings are complementary if the second, read backwards, is the same as the first, except that A and T are interchanged, and C and G are interchanged. Thus CGAATC and GATTCG are complementary. Strings that have substrings that are complementary will anneal in various ways with the complementary substrings matched. DNA strands of about 500 base pairs in length can be synthesized in the laboratory on a machine. The human genome, for comparison, contains 4 billion such base pairs.

1.4 Basic Computational Operations on a DNA Strand

The basic operations are as follows:

- DNA strands can be constructed that correspond to any desired string of the letters A, T, C, or G.

- Double strands separate into single strands when heated.
- Single strands anneal into complementary double strands when cooled.
- All the single strands, or all the double strands, can be removed from a given mixture of strands.
- The strands can be cut (using restriction enzymes or constructed chemicals) at various markers.
- The strands can be separated by length.
- Strands that have a known sequence of 15-20 base pairs anywhere along the strand can be extracted and separated from the remainder of the DNA molecules.
- It is possible to find the string that a DNA strand encodes.
- Given short (15-20 bases) known segments anywhere in a strand of DNA, the complementary strand can be constructed.
- Many copies of a strand of DNA can be constructed using the polymerase chain reaction (PCR).
- We can append a given string of DNA to a selected subset, or to all, of the DNA that is present.

Simple models for DNA computing assume that all these operations can be done without errors. In a later section we will discuss some of the practical issues involved in error correction and some engineering issues that are related to the chemical and biochemical steps of the above operations.

1.5 An Example: Breaking 56 Bit DES Using DNA

As discussed briefly above, most of the interesting proposals for using DNA amount to computing a large table, and then looking up the answer. For instance, let $f(k)$ denote encrypting a known 8-byte message using DES with key k. Then DNA computing would construct a test tube of DNA representing $\{(k, f(k))\}$ for all 256 possible values of k. Suppose one is given an intercepted message, m, that is the result of encrypting the known message with an unknown key. One can extract the DNA representing (k_m, m) from the test tube which gives the key, k_m, that was used in the encryption. The DES algorithm of Lipton uses over 100 distinct biotechnology steps and requires about 4 months of reaction time to proceed. We discuss this example computation further below.

1.5.1 Constructing All Bit Strings

If we write integers in binary, we get strings made up of 0s and 1s. We can construct all the strings of length n in parallel. To do this we pick DNA strings $B_1(0), B_1(1), \ldots B_n(0), B_n(1)$. Then the n bit number $x_1x_2 \ldots x_n$ is to be represented by the DNA strand $B_1(x_1)...B_n(x_n)$. (Some research papers stick separators in between the bits to mark the boundaries, but these can be included in the coding of the B's.) Now for each $B_i(x)$ choose some string $v_{i,xL}v_{i,xR}$. Synthesize the DNA corresponding to all these strings and pour a lot of it into a test tube. Then synthesize the DNA complementary to $v_{i+1,yL}v_{i,xR}$ for all i and all four choices of x and y from $\{0,1\}$ and pour a lot of that into the test tube. If we choose the v's properly, the second batch of

DNA patches the bits together, and then DNA polymerase will make double strands representing n bit integers. One can separate out all the DNA that is not the right length.

For this to work even in DNA mathematics, there ought to be no big hunks of v's that could come from different places, or are complementary to each other. This can be accomplished by choosing 10 or 15 bases for each v.

1.5.2 Typical "Computation" on DNA

A typical calculation works in stages, by computing intermediate bits. At some point we might have a test tube containing strings kb where k is the input, and b has intermediate bits.

Suppose the next stage is to compute 4 bits based on the 6 right hand bits of b. (In DES, for instance, there are little tables that replace 6 bits by 4.) We separate the DNA into 16 batches by matching on the bits that give each value. Thus (in the DES case) we extract the DNA ending with the bits 000011, 010111, 100001, and 110110 into one test tube. (For these 4 choices the function we are computing has the value 3.) We then cleave off the 6 old bits (assuming we don't need them any more) and append the four bits 0011 to all the DNA in the test tube. Having done the appropriate thing for all 16 test tubes we mix them together, and go on to the next step.

This technique allows us to simulate in DNA any calculation being done by a combinatorial logic circuit augmented by small ROMs.

Most proposals of this sort have been paper studies. Adelman's exper-

iment, so far the only wet computation, proceeded slightly differently. He constructed DNA representing paths through a graph that didn't repeat vertices, and used separation techniques to find the longest strand, which gave him the Hamiltonian path he was looking for.

1.6 How Much is a Mole?

For the currently proposed DNA computing methods to be applicable to a problem, one needs to have enough DNA to insure that an exhaustive search of all possibilities is present in the initial library of strands in the test tube. A base pair has a molecular weight of about 600 g/mole. Thus, 1 g of material contains 1 g $\times$ (6×10^{23} molecules/mole)/(600 g/mole of base pairs) = 1.0×10^{21} base pairs. We note that 1 g of oligonucleotide is at present beyond the state-of-the-art of biotechnology methods, and a 1 mg quantity would be considered a very large amount in currently operational biotechnology laboratories that typically use microgram quantities at most. Nevertheless, we will consider the proposal that envisions the use of even larger amounts, say 1 g (a mere factor of 10^6 over operating practice!), of DNA for computational purposes. A 1 g quantity of DNA would then contain 1×10^{21}, or about 2^{70}, bases.

Since the mode of DNA computing identified to date requires sorting through a large number of possibilities in order to identify the unique sequence that comprises the "answer" to the computation, reading the answer requires amplification of the small individual number of "correct answer" molecules into a macroscopic quantity that can be manipulated, sorted according to the computational algorithm, and subsequently identified. In

practice, although single molecule amplification has been shown under highly controlled laboratory conditions, an operational DNA computer would probably require some redundancy to insure that the molecule(s) containing the proper answer are indeed present in the flask and have not been lost by absorption to the walls of the test tube, etc. Thus, probably 10-100 molecules of each base pair code type are desired, so the 1 g would likely operationally contain only 1×10^{19} non-redundant base pairs.

The amplification step, using PCR (polymerase chain reaction), requires that the initial binding event be highly preferential to the desired sequence of bits; otherwise, molecules containing "wrong answers", i.e, undesired base pair sequences, will be amplified as well. The thermodynamics of base pair mismatching are well known for naturally occurring bases in DNA oligonucleotides. Thermodynamically A prefers to bind to T and C prefers to bind to G, but these binding pairs are only preferred over the mismatched binding pairs by 1 kcal/mole per base pair, i.e., by less than a factor of 10 in binding constant. Since individual base pair mismatches are likely, each bit must be encoded into 15-20 base pairs to insure successful discrimination and amplification using PCR. This encoding reduces the number of bits in 1 g of oligonucleotide to approximately 10^{18} nonredundant bits/g, i.e., 2^{60} bits/g.

Since the manipulations using DNA involve heating, cooling, physical separation of solutions, molecular binding events, enzymatic events, etc., not many operational steps can readily be envisioned to solve a specific problem. Perhaps 10 operations is typical per day, with a very optimistic assumption of 100/day using new automation methods and assuming that all operations are just dilution steps and not binding or cleavage (which will take much longer with reasonable amounts of enzyme on large quantities of DNA or the biological equivalent of the usual price/time tradeoffs will occur). A currently

significant quantity of DNA, 1 mg, would then have an overall computational rate of 10^{11}–10^{12} bit-ops/sec, with a hypothetical 1 g DNA computer yielding an overall rate of 10^{14}–10^{15} bit-ops/sec. Current prices lead to the estimate that the cost of this of 1 g DNA computation would exceed $10 million just for enzymes and raw materials needed to complete a computation in about one day of "operating" time (see below).

For comparison, a special purpose pipelined DES computer could be made, in current technology, in an area of silicon about .06 cm^2. Such a computer could produce 2^{26} results per second. A one centimeter chip would produce 10^9 DES results per second. Since a DES computation is about 1,000 bit operations, such a chip is not much more than a factor of 1,000 off the biological computer. We discuss a more aggressive design below.

1.7 Is There a Niche for DNA Computing?

When might DNA computing make sense? The alternatives are using conventional computers, building special purpose hardware, settling for approximate or heuristic answers, or doing without.

Yes

An optimist would point to potential factors of 100 to 100,000 advantage over conventional general purpose computers if 1 kg of DNA (a factor of 10^6 greater than state-of-the-art!) and the expectation that progress in the near future will produce algorithms for more problems people care about. This seems possible, and it is worth acting on the possibility.

No

A pessimist would argue rather differently. The following arguments are quite one-sided.

First, the problems that DNA computing appears to be able to solve include no problems (other than DES perhaps) that anyone cares about. The kind of problems one can do in combinatorial circuits aren't interesting. Even if one could do interesting NP complete problems, most applications that want solutions to NP complete problems don't need the guarantee of exact solutions. Even then, no interesting example of an NP complete problem only has 70 or 80 bits of input. (For instance, traveling salesmen problems of 70 cities can be solved exactly.)

Second, any problem that anyone cares about could be done just as well on special purpose equipment. Consider any encryption method on N bit keys. One lays out the logic on chips, unrolling the loops. (In DES, the 16 stages are all separate.) One pipelines the operations, so getting a result every 4 nanonseconds seems unchallenging. (That's the present day clock rate of Alpha chips.) That gives 2^{28} answers/second. The other 2^{N-28} have to made up as a product of the number of seconds the machine runs times the number of copies of the circuit that the machine contains. For DES with $N = 56$, 2^{14} copies and 2^{14} seconds (5 hours) does not seem so hard. That's \$5,000,000 or \$10,000,000 and 18 months (at most) to build, and fast to run. For an answer every four months, build a machine with fewer copies or that runs slower. (The point here is that DES is not really a hard problem any longer. The algorithm wasn't designed to survive to the 21st century.) Even another 8 bits of key wouldn't matter much, giving a solution in a month instead of in hours.

On the other hand, DNA computing to solve problems requiring exhaustive searches of 2^{60} bits would need about 1 g of DNA to proceed, assuming that some method to use all of the DNA efficiently were discovered. As described in the engineering estimates below, using current prices for materials, the purchase of raw materials (bases, enzymes, etc.) alone, without performing any manufacturing or engineering work for a 1 g scale DNA computation to break 56-bit DES would cost about $10,000,000. Note that there is already a significant biotechnology industrial demand for these materials, so we assume that their sales prices are reasonably reflective of the engineering and personnel costs required to produce large quantities of material. Additional bits in the DES key require more silicon space, money, and/or time on a conventional computer and also requires more material, money, and/or time on a DNA computer.

It is possible that with much lower enzyme and materials costs, DNA computers could do interesting DES sized computations with much less up front costs compared to building special purpose electronic hardware. Both types of machines fail on larger problems, for similar reasons. It is currently too expensive to utilize very large amounts of silicon area, and similarly it is too expensive to utilize very large quantities of DNA and the enzymes required for computing purposes. The only real difference is in the areal memory density from which the "baseline" starts, but on balance the tradeoff of memory vs. ops can be seen to probably be comparable for conventional silicon and DNA computing.

1.8 A Useful Research Program

With advances in both algorithms and engineering, DNA computing might have some advantages. A carefully designed, and relatively modest, research program could test this hypothesis.

1.9 Algorithms and Theory

At this point the only algorithm that does a problem of general interest is the DES decryptor. However, DNA computing has become a hot topic in the computer science and algorithms community. Generalizing from experience with other new models of computation, one can expect that the usual forces in the community will provide a thorough exploration of theoretical and algorithmic issues over the next two years. The people doing this will be people who have been doing other theoretical or algorithmic work, and their present funding sources should be adequate.

1.10 Experiment and Engineering

DNA chemistry is not the same as DNA mathematics. Each operation takes minutes or hours. Each operation makes mistakes. Some operations cannot be done on arbitrarily long pieces of DNA. These engineering considerations affect what is practical. A sensible program should be developed to

investigate the actual limitations of performing the various steps involved in DNA Computation on a significant scale.

1.10.1 PCR

PCR is critical in several operations. In particular, at the very end of the DES calculation, one tries to extract the small amount of DNA that corresponds to the one correct decryption out of 2^{56}. Sequencing the DNA to discover the key requires macroscopic amounts of DNA, so the extracted DNA has to be amplified greatly. The amplification essentially works by adding short segments (say 20 bases) that match both ends and then using *Taq* or some similar polymerase to double the amount of good DNA present. Repeat many times to get as much as needed. Note that we can know the left and right hand ends because we choose the encodings of the bits.

PCR presently has an error rate of about 1 in 20,000 bases. That is, if the DNA has N bases in it, about $N/20,000$ of the bases in the replica will be wrong. Successive stages of duplication will copy the incorrect DNA, possibly with additional errors.

This is not too bad for reading off the answer. If we have to do 40 generations of doubling on a strand that's 20,000 long, then each resulting strand will have about 40 errors. Since each bit is represented by about 20 bases, if the DNA can be sequenced at all, the errors can be easily corrected. Even if the sequencing is ambiguous at the level of bits, from other causes, most of these calculations have the property that it is trivial to check a purported answer on a conventional computer.

PCR errors are potentially more troubling if amplification is required at many places in the computation, especially early. For instance, if the left side of B_1 were incorrectly copied early in the DES computation, the amplification at the very end would fail completely. Some such errors can be guarded against, or checked for, without slowing the computation, by adding processing steps in parallel.

Doing single molecule PCR amplification in the presence of 1 kg of DNA, and the associated impurities therein, is another matter entirely. Even doing PCR on one molecule with 10^{18} other spectator DNA strands in the test tube needs some verification. Performing such a demonstration seems both prudent and interesting.

1.10.2 Extracting

The mathematics of DNA assumed that if we choose values for any number of bases, then we could extract all the DNA that had a string of bases at its left end. Unfortunately this is not true. Presently the longest patterns one can use are about 500 bases.

In the DES calculation, for example, we want to select on the final 64 bits. If there are 20 bases to encode each bit, essentially we want to extract based on a string of 1,200 bases. This would require at least three extractions, each extraction followed by cleaving the common bits off the right hand end, and then possibly amplifying the remainder and putting together the piecemeal sequence information to yield the desired full sequence information.

1.10.3 Cutting and Reconnecting DNA

At the present, DNA molecules can be cut only at certain patterns of bases. As long as that is true, one has to choose encodings so that the DNA can be cut where, and only where, the algorithm requires. Whether one can successfully perform lots and lots of cuts and lots and lots of reconnects (hybridizations), and still reuse the same enzyme preparation also needs investigation.

1.10.4 Very Large Scale Production of Restriction Enzymes, Ligases and Affinity Columns

DNA computers solving interesting problems using current algorithms will use incredibly large amounts of very expensive materials, such as ligases, restriction enzymes, affinity columns, etc. Methods to greatly lower the production costs of these materials, and to produce them in enormous quantities relative to current biotechnology applications, are required in order to envision a feasible DNA computer. An engineering program directed towards investigating the scaling of production of certain key enzymes would be essential.

1.10.5 Controlling the Yield of Desired Length DNA Strings

DNA computing is most attractive when an exhaustive search is re-

quired, because the binding of the oligonucleotides can be used to construct a library. However, the length of the strands formed during this binding needs to be controlled, and prescribed experimentally, otherwise a large fraction of the strands will be of incorrect lengths. Although strands of undesired lengths could be rejected through a sorting step based on strand length, most of the (expensive) DNA will be wasted if the strand length is not controlled experimentally during formation of the initial library. Thus, stop bits need to be encoded. Further, the yield to completion of the encoded bits to form the entire library needs to be investigated experimentally; otherwise the complete library will not be formed and additionally, DNA will be wasted. Thus, a sensible research program would include experiments to investigate the yield of binding events to reach a desired bit length in the encoded oligonucleotide strands.

1.11 Circumventing the Restriction of a Mole of DNA

One would really like to be able to solve problems without having the requirement of first preparing an initial library that contained an exhaustive search of all of the possible solutions to the problem, thereby using up the precious and costly DNA. One approach to the problem would be to use "artificial evolution" procedures, in which the DNA is mutated deliberately and the mutations amplified, transcribed into RNA, and "correct answers" selected for and reinforced by assaying some cellular level functionality. In this fashion, the initial finite-sized, and perhaps incomplete (but hopefully randomly chosen and spanning all possible answers, in this case) DNA library could be used as starting points from which the correct answer could be reached through mutations, much like approaching a local (and hopefully

global) minimum in a conventional computation from a set of starting parameters. We have no specific idea how to encode the solution to a DNA-based computational problem into a cellular-level function, and apparently neither does anyone else at present.

Good ideas and good algorithm development using this approach, should be carefully considered.

1.11.1 A Cost Comparison

It is interesting to consider the cost drivers and then to compute the cost of doing a DNA calculation on a 1 g scale (i.e., suitable to break 56 bit DES). In performing this cost estimate, we assume a high yield of strands of the desired length can be obtained; otherwise absolutely unrealistic quantities of DNA would be required to prepare 2^{56} strands of correct length. Making the assumption of unit yield for formation of strands of the correct length, using current prices in the Sigma Chemical Co. Catalog, and some reasonable estimates for the future, we describe below the costs for the various steps in the DNA computing process, using the 56 bit DES algorithm as an example:

1.11.2 Nucleotides (i.e., Raw Materials)

On p. 1464 of Sigma catalog, one can find the deoxynucleotides A, T, C, G from which we will make our DNA strands using a "gene machine". The raw materials cost \$2/micromole. The 1 g of DNA contains about 10^{-2}

moles of bases, so the current cost is $\$2 \times 10^4$ for raw materials. This expense is not the ultimate cost driver currently (vide infra), and it is our estimation that it could come down significantly if demand increased for these bases. The oligonucleotides could also be made synthetically as opposed to enzymatically, which would greatly reduce this component of the raw materials cost.

1.11.3 Ligases

Each bit in our computation is a 20-mer, and all of the bits need to be tied together after their initial binding so that the subsequent sorting and selection steps in the algorithm can be performed. Such ligation will almost always be necessary at the initial stages of any DNA computation in order to convert weak complexes into double stranded DNA that can be manipulated using standard biotechnology methods. In standard biotechnology laboratory procedures, 0.25 units/μl of ligase ties together 25 femtomoles/μl of ends in 1 hour (per G. Joyce). Using this as a guide, we wish to compute how much ligase would be needed, and how much such a quantity would cost, in order to perform this initial binding step in the DES algorithm on our 1 g of DNA in one hour.

At 1 g DNA, 600 g/mole base pair and 20-mers to be ligated, we have about 8×10^{-5} moles of ends to tie together. Since 2.5×10^{-1} units of ligase ties together 2.5×10^{-14} moles of ends, 1 unit of ligase ties together 10^{-13} moles of ends. We have to have 8×10^{-5} moles of ends to be ligated, so we require 8×10^8 units of ligase. This is an amazing amount of ligase, just to

perform the first computational step with our 1 g of DNA. At current prices of \$1/unit, this amount of ligase would cost $\$1 \times 10^9$.

To put this in perspective, we will consider producing this much ligase, and producing it ourselves by cloning a gene and fermenting cells to grow the ligase. Fermenting 1 liter of cells with a T4 ligase-closed gene can yield 10^6 units of ligase, so our 1×10^9 units requires about 1000 l of fermentation. This would be about 1 years production of a very big (U.S. Biochemicals, Inc.) house devoted entirely to this step. This clearly seems to justify a cost estimate of $\$10^6 - 10^7$ for a custom, large scale, preparation of enough ligase for the first step in the DNA computational process using current biotechnology methods.

The ligase could be reused for subsequent ligation steps (at least in principle), although stability of the enzyme in repeated steps is required over a four month period just to complete one full DES computation. Due to parasitic decomposition pathways (free radicals, etc.) this level of stability has not been achieved to date in any laboratory, but we will assume highly optimistically that it could be done, and even extended in duration, to produce 10 such complete computations, after which the ligase would need to be replaced. The materials cost of the ligase is therefore, very optimistically, about $\$10^5 - 10^6$ averaged over the lifetime of the "computer", assuming a > 3 year stability period of the initial ligase batch.

The ligation step also needs ATP to work; furthermore, the concentration of ATP should be 1 mM of ATP for the ligation process to work at the rate quoted above. The ATP concentration is typically 1 mM in standard biotechnology procedures of ligation, so the total ATP needed in our 1200 l of ligase is about 1.0 moles. ATP production doesnt scale well, and it currently

costs \$5,000/mole. This \$5,000 materials cost is probably a minimum cost for this step.

The DES algorithm contained about 10^3 individual steps requiring ligation over a 4 month period. Unlike the ligase, the ATP is consumed in each step. Fortunately, the amount of ends that are to be tied goes down by a factor of 100 after the first step, since most of the bits are already tied together and only the new sequences which are to be introduced onto the ends of the DNA in each computational step need to be formed into double-stranded DNA. Also, one might envision better algorithms that only used 100 ligation steps. Considering this favorable situation therefore adds a factor of on the order of two to the cost of the ATP, for a total ATP cost under favorable circumstances of \$10,000. For the foreseeable future, the ligase cost thus greatly dominates that of the ATP.

From these calculations, it becomes clear that the cost of the ligase step is directly proportional to the amount of DNA needed for the computation, if the computational step is to be done in a constant time. Another way of stating this is that the current cost of a ligase operation is 10^{8-9} ligase ops/(sec-\$), since 1 g of DNA contains about 1×10^{18} nonredundant bits, and we perform 1×10^{18} ops in 3.6×10^3 sec for $\$10^6 - 10^7$ in the first problem to be solved. One can pay less and wait longer or do less ops in the same time, just like the situation on any other computer.

Note also that the volume of this process is not negligibly small. The DNA concentration must be kept below an upper bound so that nonspecific binding of noncomplementary base pairs is minimized. The "standard" concentration of 2.5×10^{-14} moles of ends dissolved into 1 μl of water for a ligation step translates into a concentration of 2.5×10^{-8} moles of ends per

liter, which is 10^{-6} moles of base pairs per liter. At 600 g/mole of base pairs, this is 6×10^{-4} g base pairs/liter, so our 1 g DNA computer would require 2×10^3 liters of H_2O for the ligation step. This is a fairly impressively sized computer, as opposed to a test tube; our 1 g DNA computer would fill up a cabinet and would have a total weight of 2000 kg.

1.11.4 Restriction Enzymes

The DES algorithm used about 100 different restriction enzyme cutting steps, all of which have to be performed on the entire 1 g of ligated DNA strands. The cheapest restriction enzymes currently cost about \$0.10/unit (see the Sigma catalog p. 1434), where a unit is defined as fully cleaving 1×10^{-6} g of plasmid DNA in 1 hour, and most cost over a factor of 10 more per unit. Plasmid DNA contains 6000 base pairs in 1 g , and plasmid DNA typically has 2 cleavage sites per strand; thus, 1 g of plasmid DNA contains {(2 cleavage sites/plasmid) $\times$ (6×10^{23} base pairs/mole)}/{(600 g of base pair/mole) $\times$ (6000 base pairs/plasmid)} = 3×10^{17} cleavage sites. This means that the activity of 1 unit of restriction enzyme is about 3×10^{11} cleavages/hour, so the cost is 3×10^{12} cleavages/\$-hour for the most inexpensive enzyme. After the first ligation step in the DES algorithm, the 1 g of ligated DNA contains 60 bits encoded at 20 base pairs/bit, or (1 g) $\times$ (6×10^{23} base pairs/mole)/(600 g/mole base pair)$\times$(1200 base pairs/strand) = 8×10^{17} strands that need to be cut. Thus, at current prices, cleaving this 1 g of DNA in 1 hour would conservatively require about $\$10^5$, and for most enzymes, $\$10^6$.

Since each cleavage step in the DES algorithm must occur at a different sequences in the DNA strands, a different restriction enzyme is required for each operation. The cost of the restriction enzymes is therefore about $\$10^7$ under favorable assumptions. This cost also assumes, as was done with the ligase, that an economy of scale can be realized in custom fermentation of all of the restriction enzymes, as opposed to just the most inexpensive one, such as ECOR1. If one reduced the number of cutting steps to 16, the cost drops to $\$1.5 \times 10^6$ under favorable circumstances. One could ferment some inexpensive restriction enzymes, but one needs a great variety of restriction enzymes so that each cutting step can be performed at the desired site as dictated by the algorithm. There will be a substantial investment and/or materials cost for these steps as well, probably comparable to or exceeding that of the ligase fermentation cost.

1.11.5 Hybridization Enzymes

This cost pays for the transferase which is used to add bases onto sticky ends. Transferase now costs about \$0.1/unit (p. 1448 of the Sigma catalog), where a unit sticks 1 nmole of nucleotide per hour onto DNA in the test tube. We have 10^{-5} moles of sticky ends to deal with in our 1 g scale DES-breaking pot (assuming each strand is about 1000 base pairs in length), so we need $\$10^3$ to accomplish this operation in 1 hour. The enzyme could in principle be reused for a while, so this is a one-time investment cost like the ligase cost.

1.11.6 Polymerase

1 unit converts 10 nmoles of bases into DNA strands in 30 min; i.e., about 10^{-9} moles/hour of nucleotide is converted into DNA. Typical costs are \$0.1 to \$1/unit. There isn't, however, anticipated to be a great expense here, because the amplification is proceeding on a small, sorted set of material at the very end of the computation, and only has to be done enough times to make sufficient DNA to sequence. This is therefore not a cost-driver at present.

1.11.7 Cost Summary and Comparison

We did not estimate any of the capital costs associated with affinity separation columns, gel electrophoresis columns, DNA-making machines, etc., or costs associated with the process engineering involved in handling 1 g of DNA. Ignoring the cost of the raw bases themselves and assuming that they will come down with significant demand, the cost drivers are seen to be the restriction enzymes and the ligase. These are expensive for reasons which are similar to why it is expensive to use electron beam lithography to create increased density memories and thereby increase the computing capacity of conventional computers. Estimated raw materials costs to break a 56-bit DES in 4 months on DNA are thus very conservatively on the order of \$1,000,000 to \$10,000,000 (or more, by factors of $10^2 - 10^3$). As described above, this highly optimistic estimate, far lower than today's bulk prices for the needed raw materials and assuming a research breakthrough in complete utilization of the starting oligomers to for the desired library, is comparable

to the complete current materials and engineering cost of building a special purpose silicon computer for this purpose, with the silicon computer performing the task of breaking the 56 bit DES in less than 1 day. Both methods of computation cost less if one wishes to let the computation take longer and both methods cost more to perform in a fixed time if the key gets bigger.

1.12 Summary and Conclusions

DNA computing comprises a very interesting area of theoretical computer science. It will be prudent to encourage the computer science community to explore DNA computer algorithm development and see what arises.

It also seems prudent to establish experimentally whether some of the proposed algorithmic steps can actually be performed in real-life on reasonable scales, or whether very tiny amounts of impurities, small but finite and unanticipated error rates, nonspecific binding events, etc. will confound a straightforward transfer of "algorithms on paper" to actual DNA computational manipulations.

DNA appears to be well-suited for computations that can be programmed to utilize a low number of operations in a highly parallel fashion. It works best at present for problems where an exhaustive search is the only alternative; it does not appear to be advantageous when this is not the case. The question is can the approach be extended to "useful" problems, or not?

Although it is too early to estimate some of the large scale costs involved with building a DNA computer, no computational step is free, including those on DNA. DNA can be used to construct large libraries, but large libraries

will require large amounts of DNA. Manipulation of these large libraries will require extraordinary quantities of enzymes and other process components, and will therefore command capital expenditures in accord with the scale of the computation. The costs involved in DNA manipulations should be considered much as the costs per bit-op are considered for conventional Si computers in assessing any given computer/computation combination. Such assessments are clearly needed to decide whether a given problem is "solvable" on any type of machine for a reasonable cost and in a reasonable time period, regardless of whether the machine is to be made out of Si, DNA or other materials developed for computational purposes.

DISTRIBUTION LIST

Director of Space and SDI Programs
SAF/AQSC
1060 Air Force Pentagon
Washington, DC 20330-1060

CMDR & Program Executive Officer
U S Army/CSSD-ZA
Strategic Defense Command
PO Box 15280
Arlington, VA 22215-0150

A R P A Library
3701 North Fairfax Drive
Arlington, VA 22209-2308

Dr Arthur E Bisson
Director
Technology Directorate
Office of Naval Research
Room 407
800 N. Quincy Street
Arlington, VA 20350-1000

Dr Albert Brandenstein
Chief Scientist
Office of Nat'l Drug Control Policy
Executive Office of the President
Washington, DC 20500

Mr. Edward Brown
Assistant Director
ARPA/SISTO
3701 North Fairfax Drive
Arlington, VA 22203

Dr H Lee Buchanan, I I I
Director
ARPA/DSO
3701 North Fairfax Drive
Arlington, VA 22203-1714

Dr Ashton B Carter
Nuclear Security & Counter Proliferation
Office of the Secretary of Defense
The Pentagon, Room 4E821
Washington, DC 20301-2600

Dr Collier
Chief Scientist
U S Army Strategic Defense Command
PO Box 15280
Arlington, VA 22215-0280

D T I C [2]
Cameron Station
Alexandria, VA 22314

Mr John Darrah
Senior Scientist and Technical Advisor
HQAF SPACOM/CN
Peterson AFB, CO 80914-5001

Dr John M Deutch
Under Secretary
DOD, OUSD (Acquisition)
The Pentagon, Room 3E933
Washington, DC 20301

Dr Douglas Eardley
618 Miramonte Drive
Santa Barbara, CA 93109

Mr John N Entzminger
Chief, Advance Technology
ARPA/ASTO
3701 North Fairfax Drive
Arlington, VA 22203-1714

Mr Dan Flynn [5]
OSWR
Central Intelligence Agency
Washington, DC 20505

Dr Norval Fortson
University of Washington
Department of Physics
FM-15
Seattle, WA 98195

DISTRIBUTION LIST

Dr Richard L Garwin
IBM TJ Watson Research Cntr
P O Box 218
Route 134 & Taconic State Prkwy
Yortown Heights, NY 10598

Dr Paris Genalis
Deputy Director
OUSD(A&T)/S&TS/NW
The Pentagon, Room 3D1048
Washington, DC 20301

Dr Lawrence K. Gershwin
Central Intelligence Agency
NIC/NIO/S&T
7E47, OHB
Washington, DC 20505

Dr David A Hammer
109 Orchard Place
Ithaca, NY 14850

Mr. Thomas H Handel
Office of Naval Intelligence
The Pentagon, Room 5D660
Washington, DC 20350-2000

Dr Robert G Henderson
Director
JASON Program Office
The MITRE Corporation
7525 Colshire Drive
Mailstop Z561
McLean, VA 22102

Dr Barry Horowitz
President and Chief Exec Officer
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730-1420

Dr Paul Horowitz
Harvard University
Lyman Laboratory of Physics
Cambridge, MA 02138

Dr William E Howard I I I [2]
Director of Advanced Concepts &
Systems Design
The Pentagon Room 3E480
Washington, DC 20301-0103

Dr Gerald J Iafrate
U S Army Research Office
PO Box 12211
4330 South Miami Boulevard
Research Triangle NC 27709-2211

J A S O N Library [5]
The MITRE Corporation
Mail Stop W002
7525 Colshire Drive
McLean, VA 22102

Dr Anita Jones
Department of Defense
DOD, DDR&E
The Pentagon, Room 3E1014
Washington, DC 20301

Dr Bobby R Junker
Office of Naval Research
Code 111
800 North Quincy Street
Arlington, VA 22217

Dr Steven E Koonin
California Institute of Technology
Vice President and Provost
206-31
Pasadena, CA 91125

Lt Gen, Howard W. Leaf, (Retired)
Director, Test and Evaluation
HQ USAF/TE
1650 Air Force Pentagon
Washington, DC 20330-1650

DISTRIBUTION LIST

Dr Nathan S Lewis
California Institute of Technology
Division of Chemistry and
Chemical Engineering: 127-72
Pasadena, CA 91125

Mr. Larry Lynn
Director
ARPA/DIRO
3701 North Fairfax Drive
Arlington, VA 22203-1714

Dr. John Lyons
Director of Corporate Laboratory
US Army Laboratory Command
2800 Powder Mill Road
Adelphi, MD 20783-1145

Col Ed Mahen
ARPA/DIRO
3701 North Fairfax Drive
Arlington, VA 22203-1714

Dr. Arthur Manfredi
OSWR
Central Intelligence Agency
Washington, DC 20505

Mr James J Mattice
Deputy Asst Secretary
(Research & Engineering)
SAF/AQ
Pentagon, Room 4D-977
Washington, DC 20330-1000

Dr George Mayer
Office of Director of Defense
Reserach and Engineering
Pentagon, Room 3D375
Washington, DC 20301-3030

Dr Greg Moore [10]
Office of Research and Development
Central Intelligence Agency
Washington, DC 20505

Dr Bill Murphy
Central Intelligence Agency
ORD
Washington, DC 20505

Mr Ronald Murphy
ARPA/ASTO
3701 North Fairfax Drive
Arlington, VA 22203-1714

Dr Julian C Nall
Institute for Defense Analyses
1801 North Beauregard Street
Alexandria, VA 22311

Dr Ari Patrinos
Director
Environmental Sciences Division
ER74/GTN
US Department of Energy
Washington, DC 20585

Dr Bruce Pierce
USD(A)D S
The Pentagon, Room 3D136
Washington, DC 20301-3090

Dr William H Press
Harvard College Observatory
60 Garden Street
Cambridge, MA 02138

Mr John Rausch [2]
Division Head 06 Department
NAVOPINTCEN
4301 Suitland Road
Washington, DC 20390

Records Resource
The MITRE Corporation
Mailstop W115
7525 Colshire Drive
McLean, VA 22102

DISTRIBUTION LIST

Dr Victor H Reis
US Department of Energy
DP-1, Room 4A019
1000 Independence Ave, SW
Washington, DC 20585

Dr Fred E Saalfeld
Director
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5000

Dr Dan Schuresko
O/DDS&T
Central Intelligence Agency
Washington, DC 20505

Dr John Schuster
Technical Director of Submarine
and SSBN Security Program
Department of the Navy OP-02T
The Pentagon Room 4D534
Washington, DC 20350-2000

Dr Michael A Stroscio
US Army Research Office
P. O. Box 12211
Research Triangle NC 27709-2211

Superintendent
Code 1424
Attn Documents Librarian
Naval Postgraduate School
Monterey, CA 93943

Ambassador James Sweeney
Chief Science Advisor
USACDA
320 21st Street NW
Washington, DC 20451

Dr George W Ullrich [3]
Deputy Director
Defense Nuclear Agency
6801 Telegraph Road
Alexandria, VA 22310

Dr Walter N Warnick [25]
Acting Director for Program Analysis
U S Department of Energy
ER30 / OER
Washington, DC 20585

Dr Peter J. Weinberger
22 Clinton Avenue
Maplewood, NJ 07040

Dr Edward C Whitman
Dep Assistant Secretary of the Navy
C3I Electronic Warfare & Space
Department of the Navy
The Pentagon 4D745
Washington, DC 20350-5000

Dr Ellen D Williams
University of Maryland
Dept of Physics & Astronomy
College Park, MD 20742-4111

Capt H. A. Williams, U S N
Director Undersea Warfare Space
& Naval Warfare Sys Cmd
PD80
2451 Crystal Drive
Arlington, VA 22245-5200

Mr Charles A Zraket
Trustee
The MITRE Corporation
Mail Stop A130
202 Burlington Road
Bedford, MA 01730